nsdi '26

FENIX: Enabling In-Network DNN Inference with FPGA-Enhanced Programmable Switches

Xiangyu Gao, Tong Li, Yinchao Zhang, Xiangsheng Zeng, Su Yao, Ke Xu

Tsinghua University

May, 2026

Background

The demand for real-time network intelligence has pushed ML inference from the control plane into the data plane.

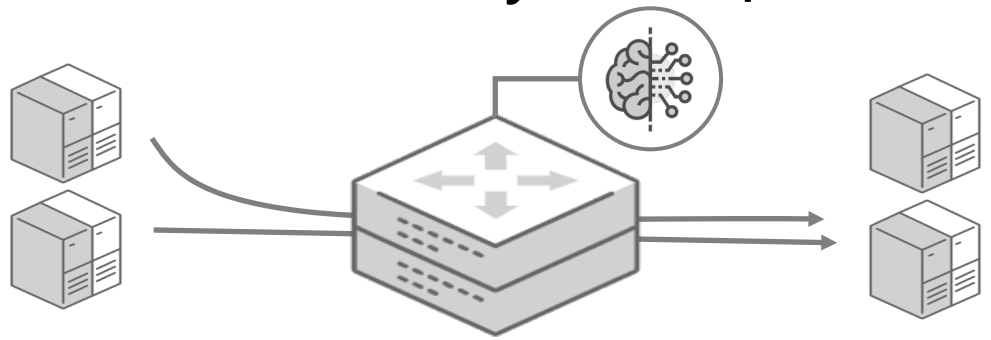
Past : Control Plane ML

High Latency (>1000μs)

Early designs placed ML in the control plane, leveraging existing infrastructure but suffering from high latencies, which is too slow for time-sensitive tasks like intrusion detection.

Present : Data Plane ML

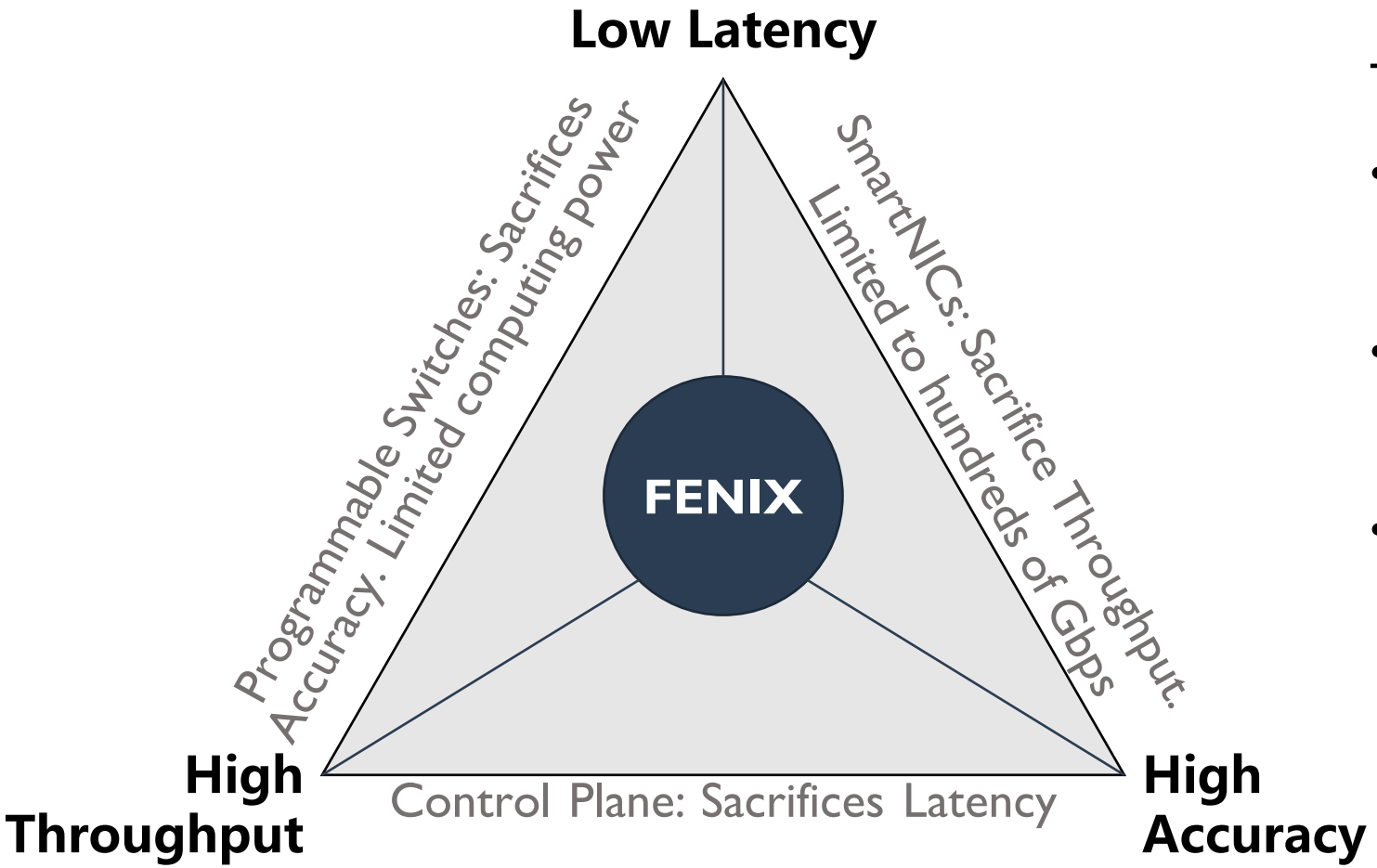
Low Latency (<100μs)



To reduce latency, modern approaches move ML inference directly into the data plane.

Motivation

Current data plane solutions force a trade-off between **latency, throughput, and accuracy**, creating an unresolved trilemma.

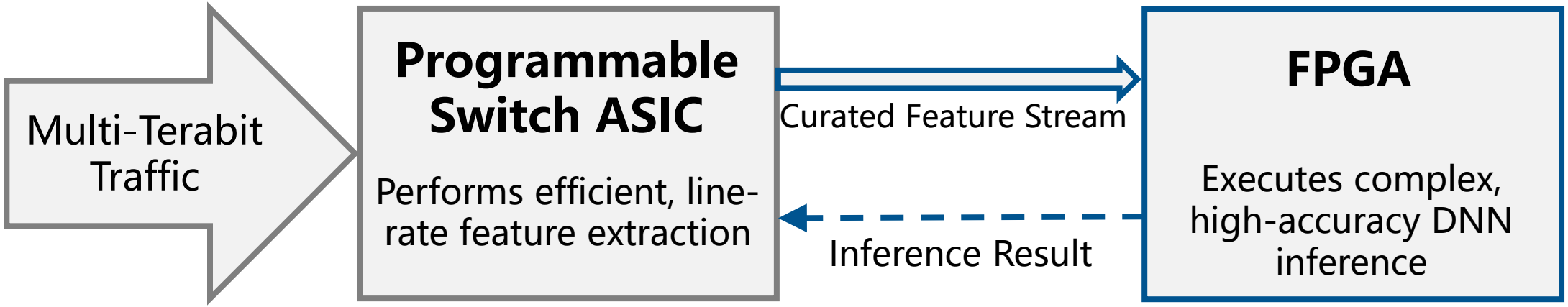


The Trilemma:

- **Low Latency:** Required for real-time response.
- **High Throughput:** Must match multi-terabit core network speeds.
- **High Accuracy:** Requires complex DNN models that current hardware cannot support directly.

Motivation

FENIX resolves the trilemma with a hybrid architecture combining a switch ASIC **Data Engine** and an FPGA **Model Engine**.



Low Latency

Microsecond-level inference with direct hardware interfacing, no software stacks.

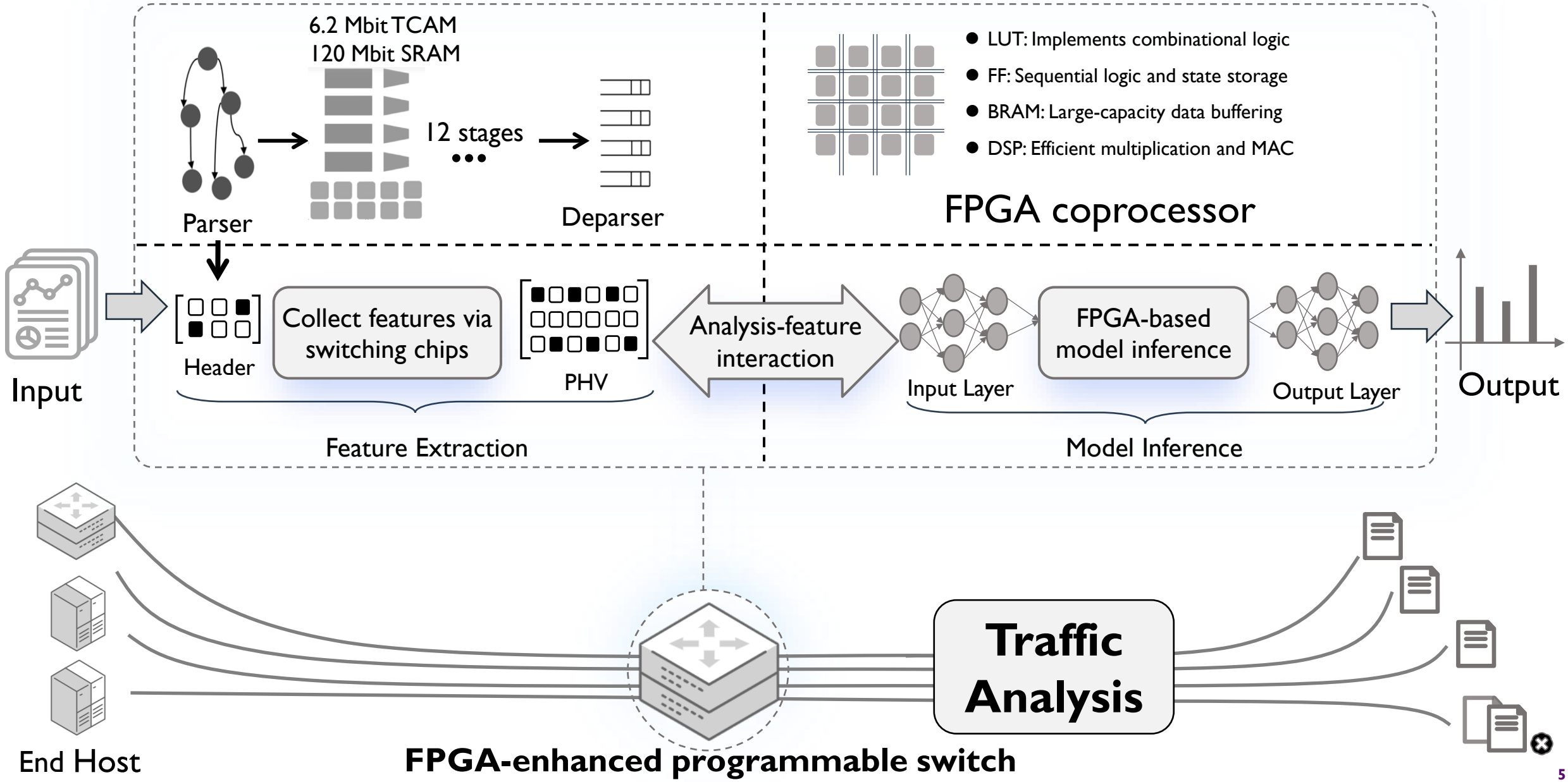
High Throughput

Line-rate feature extraction on the switch, with optimized communication to the FPGA.

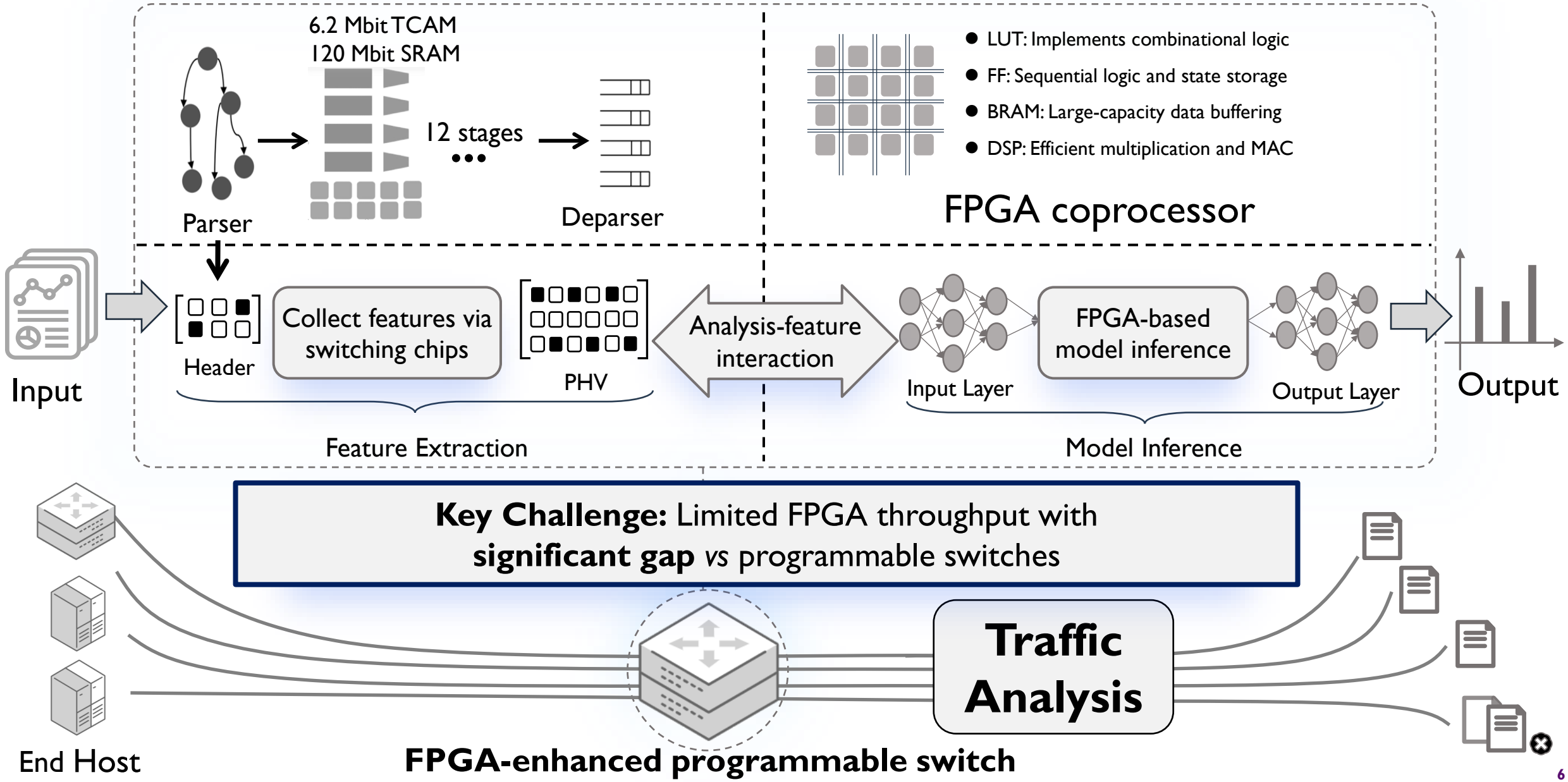
High Accuracy

Support for full DNN models (CNN, RNN) with minimal quantization loss.

Challenge

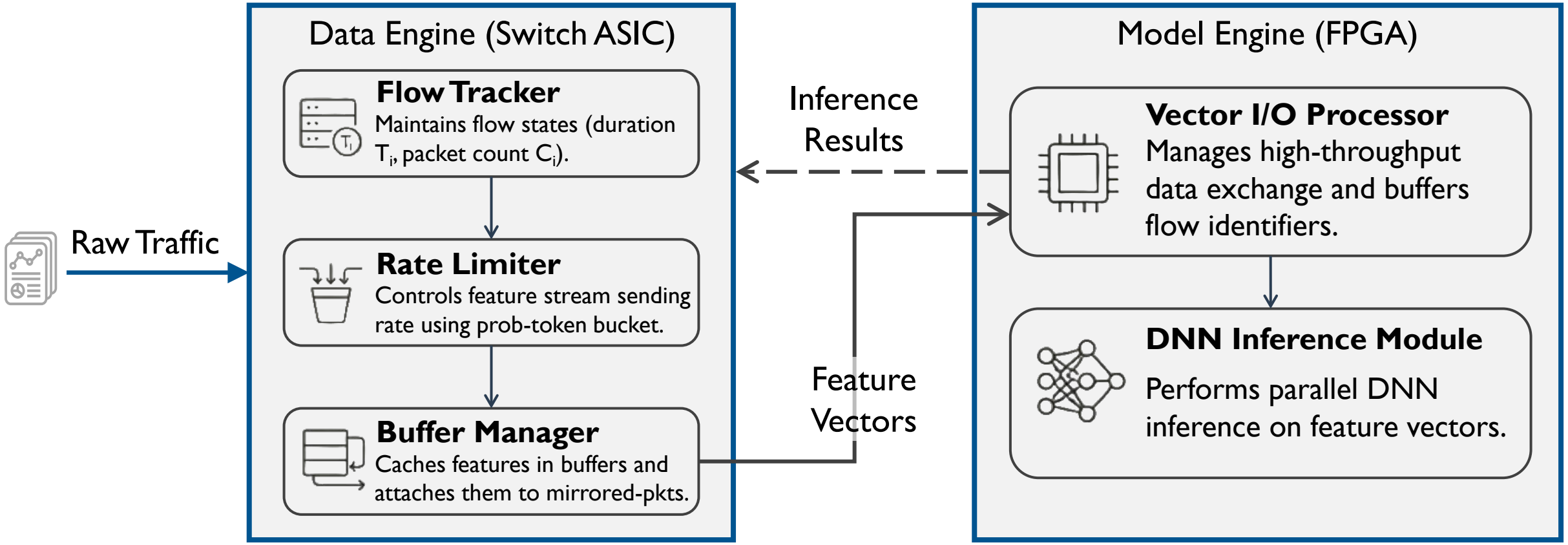

6.2 Mbit TCAM
120 Mbit SRAM
12 stages
Parser
Deparser
LUT: Implements combinational logic
FF: Sequential logic and state storage
BRAM: Large-capacity data buffering
DSP: Efficient multiplication and MAC
FPGA coprocessor
Input
Header
Collect features via switching chips
PHV
Analysis-feature interaction
Input Layer
FPGA-based model inference
Output Layer
Output
Feature Extraction
Model Inference
End Host
FPGA-enhanced programmable switch
Traffic Analysis

Challenge

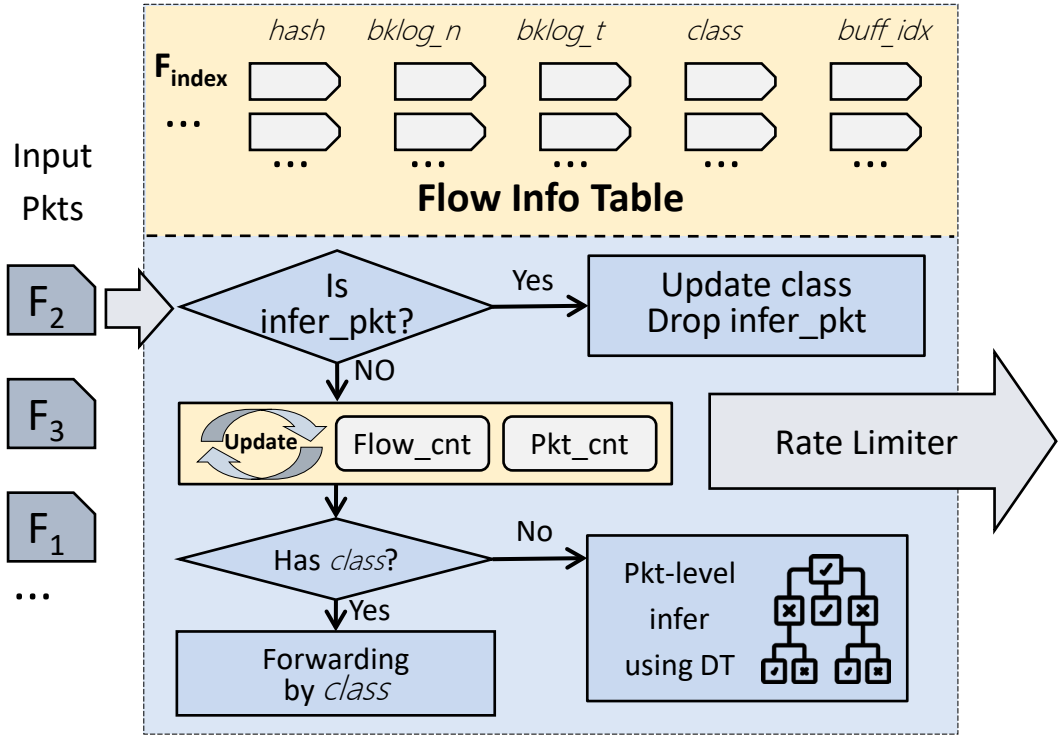

6.2 Mbit TCAM
120 Mbit SRAM
12 stages
Parser
Deparser
LUT: Implements combinational logic
FF: Sequential logic and state storage
BRAM: Large-capacity data buffering
DSP: Efficient multiplication and MAC
FPGA coprocessor
Input
Header
Collect features via switching chips
PHV
Analysis-feature interaction
Input Layer
FPGA-based model inference
Output Layer
Output
Feature Extraction
Model Inference
Key Challenge: Limited FPGA throughput with significant gap vs programmable switches
Traffic Analysis
End Host
FPGA-enhanced programmable switch

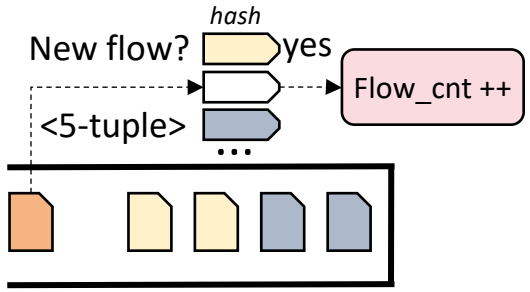
Design: Overview

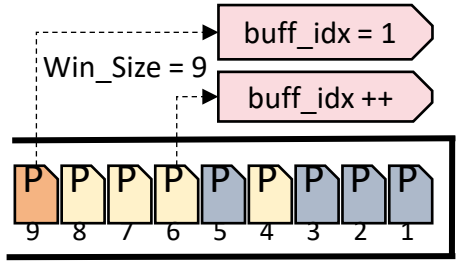
The FENIX (**F**PGA **E**nabled **N**eural **I**nference e**X**ecution for network switches) is composed of Data Engine on switch ASIC and Model Engine on FPGA.



Design: Data Enigne

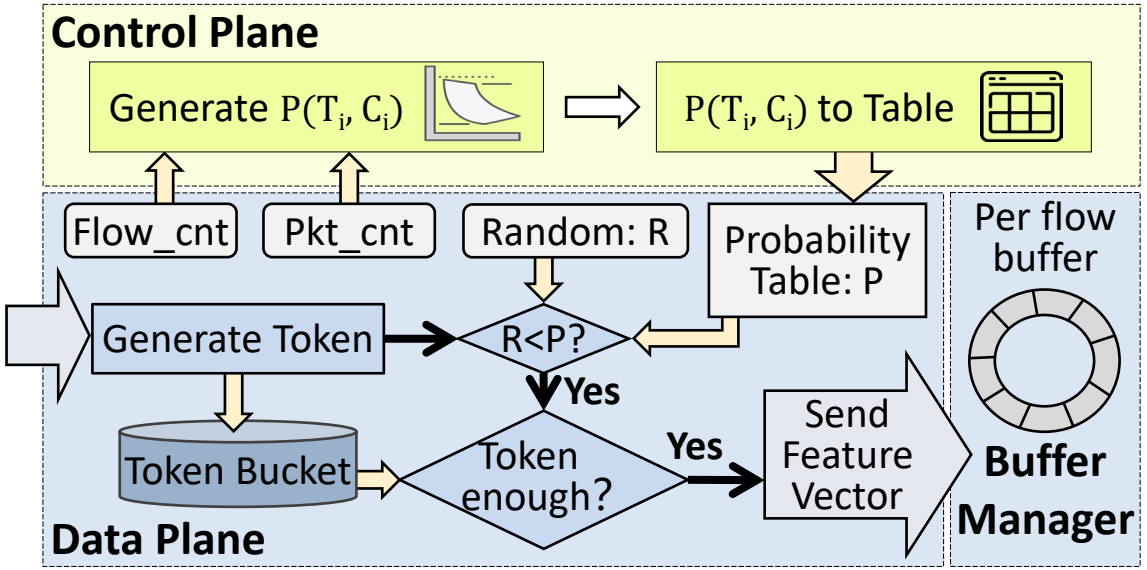


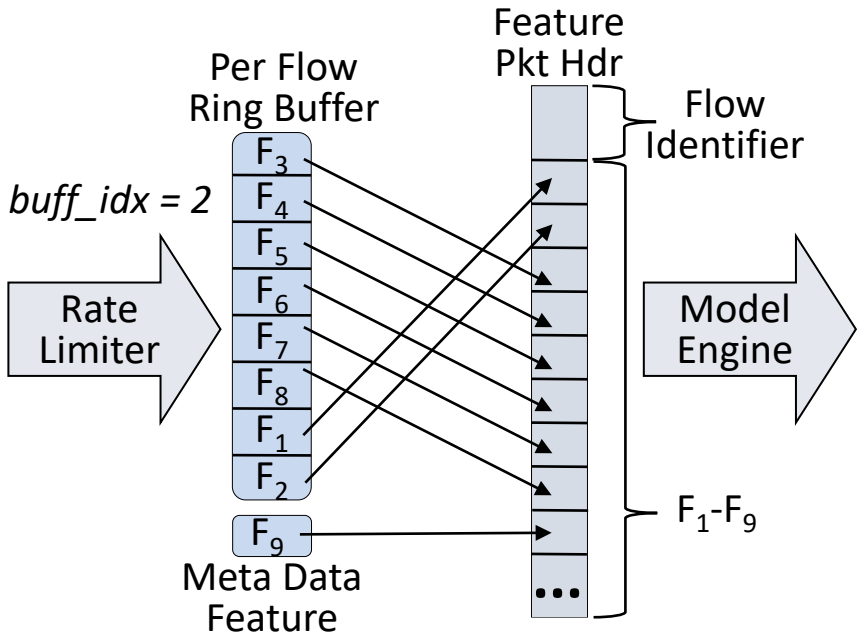




- Flow Tracker is the core of the Data Engine, maintaining a flow info table in switch SRAM indexed by 5-tuple hashes.
- Each entry stores the flow hash, backlog packet count and timestamp, classification result, and buffer index.
- On packet arrival, the Flow Tracker initializes or updates the entry; classified flows are forwarded by their class, while unclassified flows are temporarily handled by an on-switch lightweight decision tree.

Design: Data Enigne

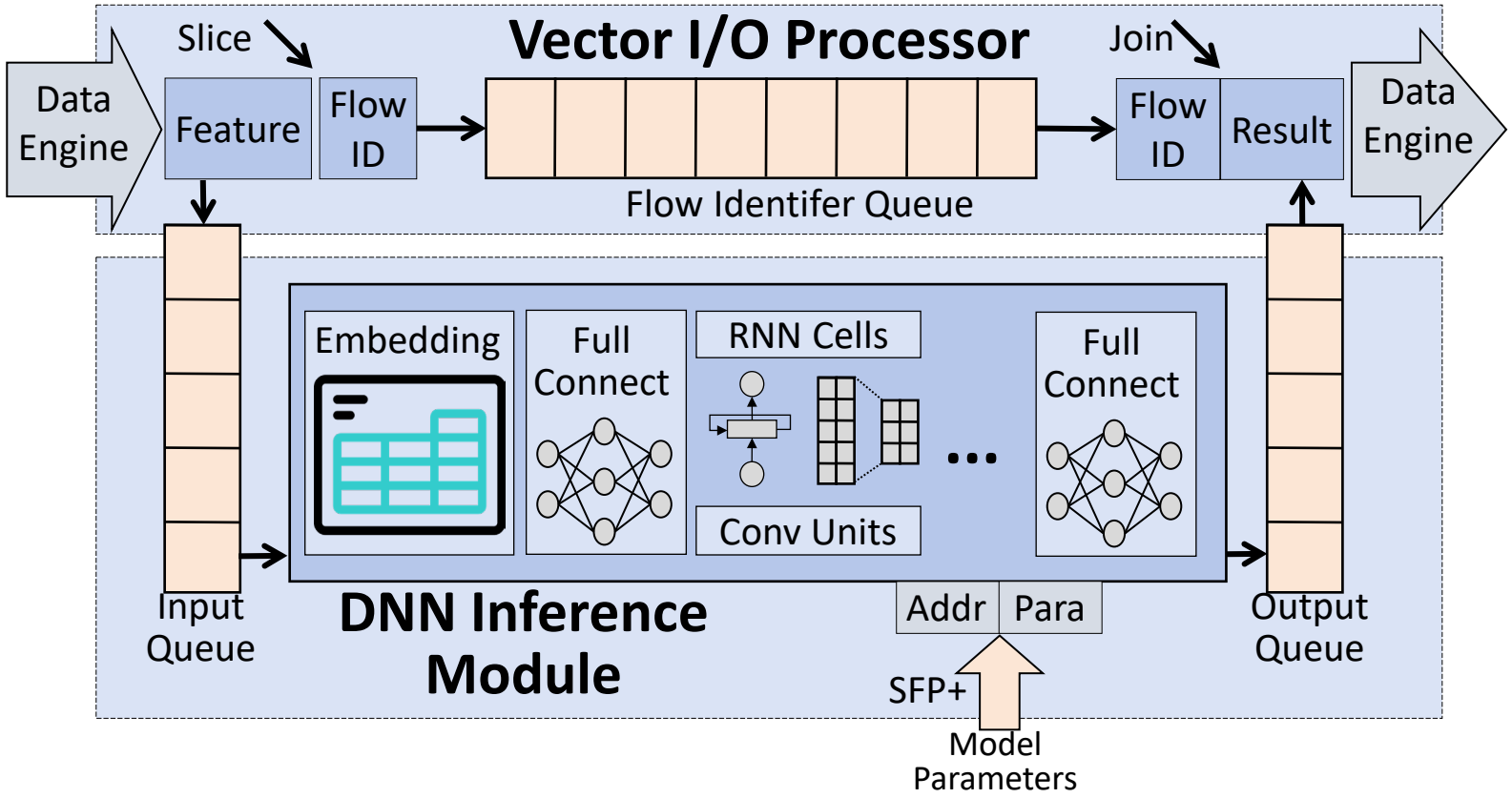




Rate Limiter & Buffer Manager

- The control plane computes the probability function P(Ti, Ci) and installs it as a lookup table in the data plane.
- In the data plane, the rate limiter uses flow/packet counters, a random value, and the token bucket to decide whether to send a feature vector.
- The buffer manager maintains a per-flow ring buffer and, when triggered, assembles buffered features (F_1–F_9) and the flow identifier into a feature packet header for the Model Engine.

Design: Model Enigne



Model Engine (FPGA)

- Vector I/O Processor splits packets into flow IDs and feature vectors, buffering flow IDs to preserve order.
- Inference results are paired with the queued flow IDs and sent back to the Data Engine for flow-level actions.

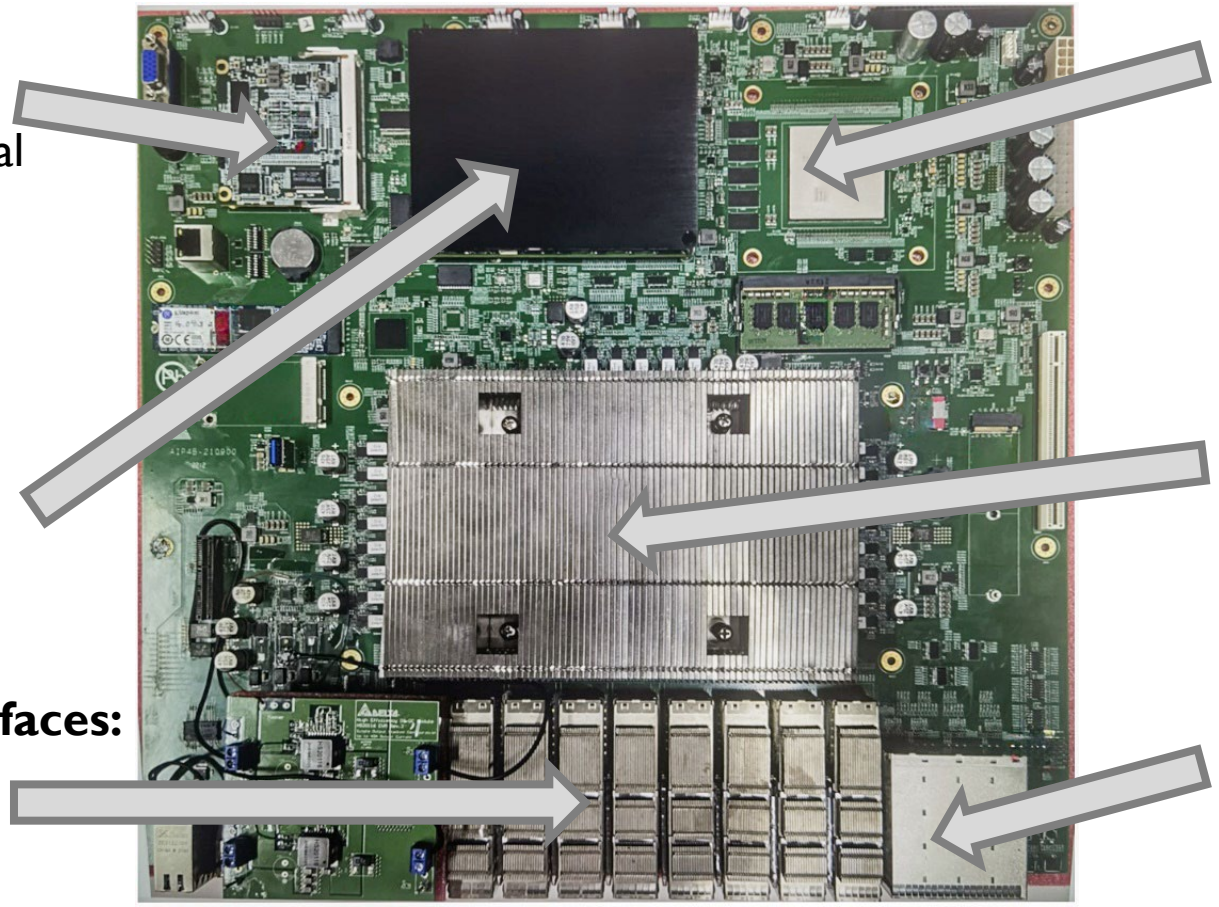
Key Feature: This design supports complex models (CNNs,RNNs) with offline INT8 quantization, preserving accuracy far beyond the binarized or simplified models possible on a switch ASIC alone.

Implementation

BMC module:
Used for hardware status monitoring, including thermal control.

Control Plane CPU:
Manages switch configuration and probability model updates.

QSFP28/QSFP-DD Interfaces:
Supports 100G and 400G connections to the Tofino.



Xilinx ZU19EG FPGA:
1.1M logic cells,
80 Mbits on-chip memory.
Implements Model Engine.

Tofino 2 ASIC:
Responsible for pkt forward.
Implements Data Engine.

SFP+/SFP28 Interfaces:
Supports 10G and 25G connections to the FPGA.

The Tofino switch chip and the FPGA are connected via three 100 Gbps on-board channels.

Evaluation: Setup

Metrics

- Macro-F1, Precision and Recall

Tasks

- Encrypted Traffic Classification on VPN
- Malware Identification

Baseline

- Flowlens [1]
- Netbeacon [2]
- Leo [3]
- Brain-on-Switch [4]
- N3IC [5]

[1] Diogo Barradas, Nuno Santos, Luís Rodrigues, Salvatore Signorello, Fernando M. V. Ramos, and André Madeira. Flowlens: Enabling efficient flow classification for ML-based network security applications. In Network and Distributed System Security Symposium (NDSS), 2021.
[2] Guangmeng Zhou, Zhuotao Liu, Chuanpu Fu, Qi Li, and Ke Xu. An efficient design of intelligent network data plane. In USENIX Security Symposium (USENIX Security), pages 6203–6220, 2023.
[3] Syed Usman Jafri, Sanjay Rao, Vishal Shrivastav, and Mohit Tawarmalani. Leo: Online ML-based traffic classification at multi-terabit line rate. In USENIX Symposium on Networked Systems Design and Implementation (NSDI), pages 1573–1591, 2024.
[4] Jinzhu Yan, Haotian Xu, Zhuotao Liu, Qi Li, Ke Xu, Mingwei Xu, and Jianping Wu. Brain-on-switch: Towards advanced intelligent network data plane via NN-driven traffic analysis at line-speed. In USENIX Symposium on Networked Systems Design and Implementation (NSDI), pages 419–440, 2024.
[5] Giuseppe Siracusano, Salvator Galea, Davide Sanvito, Mohammad Malekzadeh, Gianni Antichi, Paolo Costa, Hamed Haddadi, and Roberto Bifulco. Re-architecting traffic analysis with neural network interface cards. In USENIX Symposium on Networked Systems Design and Implementation (NSDI), pages 513–533, 2022.

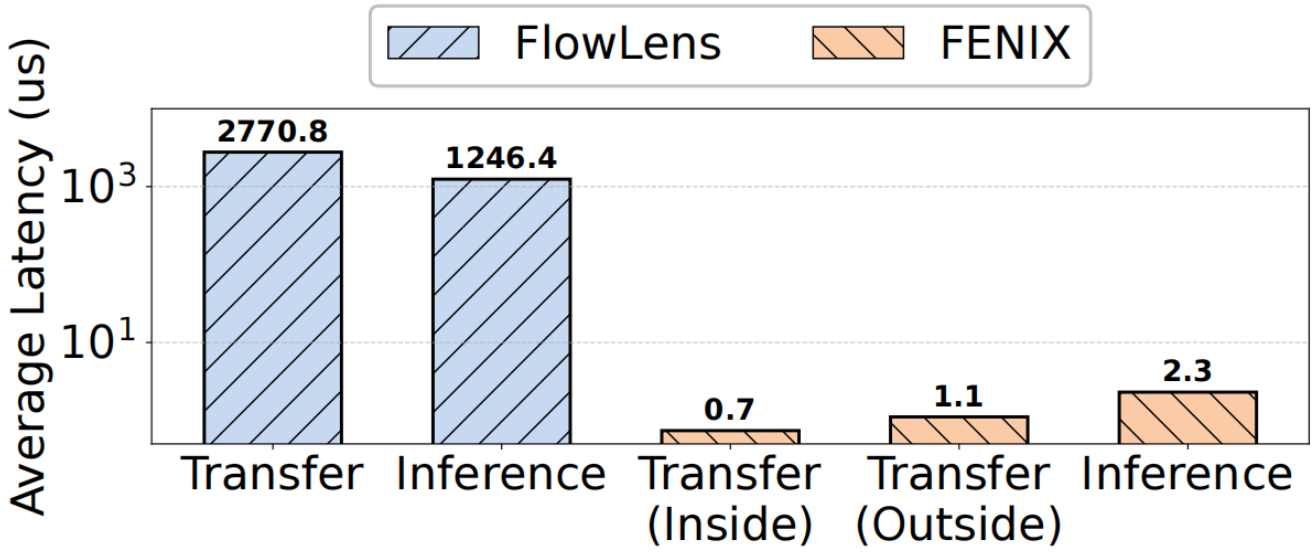
Evaluation: Accuracy Comparison

High Accuracy

Class	FENIX F-CNN	FENIX F-RNN	FlowLens [12]	FENIX P-CNN	FENIX P-RNN	NetBeacon [71]	Leo [33]	BoS [64]	N3IC [50]
Encrypted Traffic Classification (ISCXVPN2016 [25])									
Chat	0.883/0.852	0.939/0.882	0.862/0.922	0.917/0.836	0.804/0.653	0.627/0.169	0.489/0.353	0.922/0.913	0.533/0.527
Email	0.924/0.834	0.944/0.924	0.888/0.821	0.862/0.882	0.893/0.746	0.230/0.321	0.333/0.019	0.932/0.925	0.349/0.353
File	0.879/0.849	0.923/0.912	0.860/0.889	0.886/0.797	0.889/0.844	0.861/0.701	0.815/0.720	0.915/0.922	0.820/0.848
P2P	0.977/0.963	0.976/0.988	0.923/0.913	0.911/0.914	0.932/0.947	0.861/0.908	0.853/0.867	0.925/0.917	0.905/0.892
Stream	0.902/0.968	0.919/0.973	0.966/0.959	0.877/0.965	0.894/0.969	0.890/0.976	0.850/0.944	0.916/0.908	0.886/0.938
Voip	0.989/0.995	0.992/0.996	0.998/0.995	0.999/0.998	0.999/0.999	0.986/0.993	0.994/0.997	0.829/0.723	0.994/0.993
Web	0.803/0.662	0.793/0.625	0.700/0.525	0.800/0.861	0.869/0.821	0.860/0.405	0.784/0.046	0.729/0.623	0.856/0.524
Macro-F1	0.890	0.912	0.870	0.892	0.873	0.658	0.578	0.863	0.738
Malware Detection (USTC-TFC [57])									
Cridex	0.999/1.000	0.999/1.000	0.999/0.998	0.999/1.000	0.996/1.000	0.933/0.997	0.984/0.995	0.983/0.996	0.866/0.861
FTP	0.999/0.998	0.999/0.998	0.999/0.998	0.997/1.000	0.993/1.000	0.993/0.485	0.928/0.986	0.986/0.791	0.848/0.853
Geodo	0.979/0.881	0.984/0.891	0.945/0.905	0.932/0.343	0.786/0.424	0.899/0.524	0.369/0.188	0.934/0.615	0.857/0.851
Htbot	0.962/0.987	0.959/0.989	0.964/0.984	0.932/0.986	0.938/0.982	0.830/0.782	0.897/0.935	0.919/0.937	0.871/0.867
Neris	0.921/0.594	0.888/0.732	0.902/0.817	0.766/0.473	0.736/0.575	0.665/0.469	0.584/0.453	0.761/0.610	0.852/0.847
Nsis-ay	0.985/0.970	0.971/0.982	0.985/0.986	0.959/0.968	0.962/0.972	0.984/0.912	0.918/0.960	0.963/0.968	0.860/0.855
Warcraft	0.998/0.999	0.996/0.994	0.995/0.993	1.000/1.000	0.999/1.000	0.890/1.000	0.995/0.997	0.956/0.998	0.855/0.859
Zeus	0.932/0.945	0.955/0.863	0.971/0.932	0.962/0.978	0.978/0.958	0.823/0.260	0.895/0.791	0.976/0.951	0.865/0.860
Virut	0.671/0.941	0.760/0.893	0.827/0.908	0.723/0.863	0.763/0.836	0.571/0.815	0.691/0.692	0.744/0.844	0.859/0.855
Weibo	0.684/0.822	0.702/0.927	0.745/0.804	0.655/0.821	0.685/0.817	0.649/0.789	0.653/0.708	0.652/0.837	0.862/0.859
Shifu	0.999/0.966	0.997/0.995	0.982/0.914	0.947/0.829	0.965/0.767	0.194/0.302	0.610/0.593	0.835/0.501	0.862/0.859
SMB	0.700/0.522	0.845/0.504	0.726/0.654	0.632/0.416	0.665/0.492	0.607/0.434	0.555/0.491	0.651/0.406	0.862/0.859
Macro-F1	0.887	0.901	0.914	0.907	0.838	0.670	0.741	0.814	0.858

FENIX achieves over 90% classification accuracy on challenging traffic analysis tasks, delivering a superior Macro-F1 score of 0.907 on malware detection.

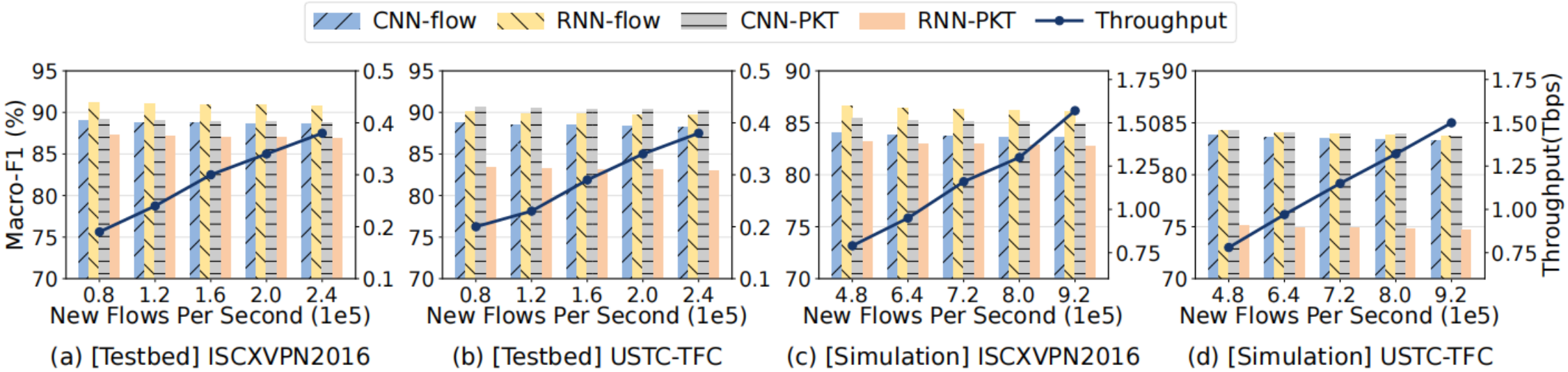
Achieve Low Latency



FENIX's tightly integrated FPGA architecture eliminates software overhead,

achieving μs-level inference latency which is up to 537× lower than control plane approaches.

Support High Throughput



FENIX leverages a probabilistic rate control mechanism to

bridge the throughput gap between multi-terabit switch ASICs and FPGAs,

enabling Tbps-level operation while maintaining about 85% accuracy at 1.5 Tbps.

Evaluation: Resource Utilization

Hardware Resource Utilization

System	SRAM	TCAM	Bus	Stage
FENIX	12.9%	4.4%	3.5%	9
FlowLens [12]	34.2%	0.0%	2.4%	9
BoS [64]	26.3%	6.3%	8.6%	12
Leo [33]	26.9%	9.0%	5.2%	12
NetBeacon [71]	11.6%	18.8%	6.4%	12

Module	LUT	FF	BRAM	DSP
CNN (overall)	38.4%	33.8%	7.1%	8.1%
Embedding	4.2%	5.1%	0.5%	0.0%
Convolutional	25.6%	19.7%	4.0%	5.7%
FC	8.6%	9.0%	2.6%	2.4%
RNN (overall)	25.6%	31.2%	6.3%	4.6%
Embedding	4.2%	5.1%	0.5%	0.0%
Recurrent	15.8%	18.7%	3.6%	2.4%
FC	8.6%	9.2%	2.2%	2.2%
Vector I/O	6.0%	4.8%	0.3%	0.0%

FENIX's hybrid architecture has low hardware overhead: on the Tofino switch ASIC, SRAM and TCAM usage are 12.9% and 4.4%, and on the FPGA, the core CNN module uses 38.4% of LUTs and 33.8% of FFs, leaving ample room for further optimization and deployment.

Conclusion

- FENIX offloads feature extraction to programmable switch ASICs and delegates DNN inference to FPGAs, using fine-grained feature rate control to **bridge the throughput mismatch** between the two types of chips.
- Experimental results show that FENIX achieves **low latency, high throughput, and high accuracy** for in-network DNN inference, providing a **practical design paradigm for future work**.

Source code: https://github.com/IntelliSwitch/FENIX

Email: gao-xy24@mails.tsinghua.edu.cn

Thanks! Questions?